Remarks

In view of the following remarks responsive to the Final Office Action dated May 21, 2007, Applicant respectfully requests favorable reconsideration of this application.

Applicant respectfully thanks the Office for withdrawing the previous rejections of the claims in view of Applicant's previous arguments. However, the Office has found and asserted new prior art and continues to reject all claims, claims 1-21, in view of this new prior art. Particularly, the Office has rejected claims 1-21 as obvious over Arora in view of Lewis.

Arora Does Not Qualify as Prior Art to the Present Application

Applicant respectfully traverses insofar as Arora is not prior art to the present application under any section of the patent statute. Particularly, the present application has a filing date of March 1, 2002. Arora has an effective filing date of December 13, 2002. Arora does, however, claim priority to provisional application No. 60/341,129 filed on December 13, 2001.

From the dates given above, the filing date of Arora is after the filing date of the present application. Therefore, it cannot constitute prior art per se. However, if the provisional application to which it claims priority contained the same subject matter of Arora upon which the Office is relying, then Arora could conceivably constitute prior art under 35 U.S.C. 102 (e).

That is not the case. Applicant has reviewed the provisional application, which is available through the Public PAIR portal of the USPTO website, and has

found that it does not contain the disclosure upon which the Office is relying in

Arora 2003/0195780.

Particularly, the Office relies on paragraphs 48 and 51 of Arora, which

discuss Figure 2 of Arora.

However, review of that provisional application, a copy of which is

attached hereto for the Office's reference, will show that the figure discussed in

paragraphs 48 and 51 of the Arora published application, Figure 2, does not

appear in the provisional application. Furthermore, neither do paragraphs 48 and

51 (or anything reasonably similar).

Arora Does Not Disclose the Present Invention In Any Event

Furthermore, Arora appears, for all intents and purposes, to be largely

irrelevant in any event. Particularly, the present invention is a technique for

running a plurality of queries to determine the location of a fixed asset. Element

(1) of claim 1 recites "detecting when a capitalized fix asset is involved in a

transaction". The Office asserted that this is disclosed in paragraph 48 of Arora

in which the economic database 250 includes fixed assets to be included in the

"what if/optimization" scenario (transaction). Thus, even by the Office's own

assertions, paragraph 48 merely discloses the existence of a fixed asset. It does

not disclose, however, what is recited in claim 1, which is a step of detecting

when such an asset is involved in a transaction. In fact, Arora clearly has

nothing to do with this. Arora, by its own description, is a system for determining

the implications of various possible business arrangements. In essence, it is a

program for running a bunch of hypothetical transactions (not even actual

transactions) to determine the business implications of various ways that

business deals and business structures can be organized. Thus, almost by

definition, Arora does not detect a transaction because, in Arora, the program is

run prior to any transaction so that the parties can determine exactly what

transaction they want to conduct.

Accordingly, contrary to the Office's assertion, even if Arora was prior art,

it does not disclose step 1 of claim 1.

Step 2 of claim 2 recites "responsive to such a detection in step (1),

running data for said asset through a plurality of queries, each query designed to

determine if said asset meets a set of criteria indicative of a category of how a

location of said asset for tax and/or insurance reporting purposes may be

determined".

The Office asserted that this is found in paragraph 51 of Arora. The Office

asserted that paragraph 51 shows:

> providing data elements, which describe things like legal entities, tax rules
> and jurisdictions that are to be modeled. (The system requires certain
> data, which qualifies as queries.) Paragraph 48 shows a regulatory
> database 248 containing jurisdiction tax laws. (Each tax law has criteria
> as to what is taxable. The jurisdiction is the location and the category is
> the type of asset taxable.)

Applicant has reviewed the Office's discussion of paragraph 51 of Arora

and step 2 of claim 1, and it does not appear that paragraph 51 (or even the

Office's description of paragraph 51) discusses anything remotely resembling

step (2) of claim 1. First, paragraph 51 does not discuss a plurality of queries,

each designed to determine if the asset meets a set of criteria indicative of a category of how a location of said asset...may be determined. Second, as noted above, there is nothing like step (1) of claim 1 in Arora in response to which the plurality of queries are run, as claimed in step (2).

Step (3) recites "if, in step (2), said asset meets said set of criteria corresponding to one of said queries, running data corresponding to said asset through an audit customized to said corresponding category to determine a location of said asset for tax and/or insurance reporting purposes".

The Office asserts that this is found in paragraph 4 of Arora, which discloses an example in which Arora weighs allocations in Nevada versus California, and asserts that this constitutes running data to determine a location.

Nothing could be further from the truth. Paragraph 4 of Arora (which <u>does</u> appear in the provisional application) discloses:

> "Other considerations, such as time-to-market, inventory, insurance, management motivation, bookkeeping, public reporting, etc., can intertwine with legal concerns in making a business decision. For example, a company with subsidiaries in California and Nevada can incur costs due to shared operations. The costs can be allocated, in varying degree, to either the California or Nevada subsidiary. In deciding how much cost to allocate a manager might realize tax advantages in California. However, another concern is that Nevada operational managers will have little incentive to conserve on costs if a large portion of the costs are being assigned to the California subsidiary."

This paragraph concerns comparing the business implications of allocating a resource to California or Nevada in order to help decide where to locate the asset. This has nothing to do with determining the <u>actual</u> location of an asset. Applicant respectfully asks the Examiner to consider the following

questions; (1) What is the asset in paragraph 4?; (2) What part of paragraph 4 discusses determining the location of this alleged asset?; and (3) What was the criteria that the asset met in order to initiate the audit described in step three of claim 1? There do not appear to be any answers to any of these questions in Arora.

In essence, even if it were prior art, Arora does not teach any of the things for which it has been cited in any event.

Conclusion

In view of the foregoing amendments and remarks, Applicant asserts that the pending claims are in condition for allowance and respectfully request that the Office issue a Notice of Allowance at the earliest possible date. The Office is invited to contact Applicant's undersigned counsel by telephone call in order to further the prosecution of this case in any way.

Respectfully submitted,

  July 27, 2007   /Theodore Naccarella/
                                              Theodore Naccarella, Reg. No. 33,023
                                              Synnestvedt & Lechner LLP
                                              1101 Market Street; Suite 2600
                                              Philadelphia, PA 19107-2950
                                              Tele: (215) 923-4466
                                              Fax: (215) 923-2189

TXN:pmf

S:\I\BM\IBM Raleigh RSW\Patents\P25673 USA\PTO\07-24-07 Response to Final 5-21.doc

PROVISIONAL

PATENT APPLICATION

COMPUTER-BASED OPTIMIZATION SYSTEM FOR FINANCIALS MANAGEMENT

Inventor(s):   Arti Arora, a citizen of India, residing at
1221 St. Matthew Way
Los Altos, CA 94024

Edward Lazear, a citizen of The United States, residing at
277 Old Spanish Trail
Portola Valley, CA 94028

Walter Hallerstein, a citizen of The United States, residing at
239 Westview Drive
Athens, GA 30606

Assignee: Liquid Engines, Inc.
1289 Hammerwood
Sunnyvale, CA 94089

Entity: Small business concern

COMPUTER-BASED OPTIMIZATION SYSTEM FOR FINANCIALS MANAGEMENT

BACKGROUND OF THE INVENTION

[01]   This invention relates in general to digital processing and more specifically to a system for analyzing allocation of factors relevant to business operations.

[02]   Maximizing performance and profitability is critical for businesses. However, this goal requires understanding complex interactions and tradeoffs that result from allocation of factors, such as cost and revenues, within a business. Businesses that have different geographic locations are subjected to different laws in different locations. Laws are often highly detailed and can implicate many characteristics of business operation such as transfer and handling of goods, tariffs, taxes, environmental effects, securities-related actions, management actions, compensation, and others. Even within a single legal jurisdiction, different laws and regulations may come into play to affect profitability and performance depending on the company's actions.

[03]   Other considerations, such as time-to-market, inventory, insurance, management motivation, bookkeeping, public reporting, etc., can intertwine with legal concerns in making a business decision. For example, a company with subsidiaries in California and Nevada can incur costs due to shared operations. The costs can be allocated, in varying degree, to either the California or Nevada subsidiary. In deciding how much cost to allocate a manager might realize tax advantages in California. However, another concern is that Nevada operational managers will have little incentive to conserve on costs if a large portion of the costs are being assigned to the California subsidiary.

[04]   This example suggests that some sort of a tradeoff analysis would be helpful in order to optimize business operations. However, such analysis is difficult due to the tax implications, alone. The incentive aspect requires measuring motivation and performance in such a way that the measurements can be compared or computed with tax benefit. Such measurements are difficult to obtain and use in a meaningful way with other financial data. Also, other factors may be involved that futher complicate analysis beyond any reasonable solution.

[05]   Traditionally, business financial decisions are made with piecemeal analysis. Statistics and reports might be used to provide some indication of an action's outcome.

1

However, such reports are backward-looking as they are based on data of past performance and can require approximation and guesswork to achieve an unreliable projection. Often the tools that are used by today's managers do not provide detailed and current calculations for tax implications, or other fast-changing and esoteric aspects of business. For large companies with a presence in many regions, analysis of even one factor can be confounding due to the many different tax laws.

SUMMARY OF THE INVENTION

[06] The present invention provides a system that analyzes, and optimizes, allocation of factors among different business entities. The analysis, or optimization, can be targeted to a specific business financials measurement, or metric. Many different types of factors can be considered regardless of their initial definition, description or form. For example, along with revenues, profit, inventory size, etc., human characteristics such as incentive and performance can be measured and optimized.

[07] A preferred embodiment of the invention concerns optimizing factor allocation to reduce overall taxes for a company with subsidiaries in regions with different tax laws. The system includes consideration of local, state, federal and international taxes, transfer pricing, tax credit limitations, inter-state allocations in unitary and non-unitary environments, carry-overs, and others.

[08] The system preferably uses a versatile matching engine, described herein, to automate the analysis and optimization. The matching engine is capable of discrete and continuous attribute value weighting, and of performing substitution of attributes, or other variables. The engine may couple a hedonic approach with linear and non-linear programming methods to obtain solutions to the optimization problem. However, many types of automated approaches can be used, such as suitably programmed general computer systems, or other combinations of hardware and software.

[09] One embodiment of the invention provides a method for assigning a financial factor to one of a plurality of business entities. The method includes steps of identifying a factor to be assigned; indicating a metric to be optimized; defining one or more rules to be used in a matching process; using a computer system to perform a step of comparing degrees of optimization of the metric based on assignment of the factor to different ones of the plurality of business entities, wherein the one or more rules are used to derive the degrees of optimization; and using the result of the comparing step to assign the financial factor to the one of a plurality of business entities.

2

[10]   In a typical embodiment, there are at least two ways to assign financial factors. The first is to assign factors, such as costs, to business entities, such as subsidiaries directly in order to maximize some objective function. An example is the assignment of costs to subsidiaries directly to minimize the tax bill (or maximize the negative value of the tax bill). After this is done, the solution can be combined in a linear or non-linear fashion with the output of maximizing another objective function, e.g., the best provision of incentives, to get a final allocation of factors, in this case costs.

[11]   The second method allows the engine to compute weights and assign them to drivers that then allocate the financial factors. For example, wage bill, square footage, and sales of each subsidiary may be defined as drivers. The optimization engine described in the previous paragraph is then used to calculate optimal weights for each of the drivers so as to maximize an objective function. Now, weights are the output of the problem and they can be combined with weights obtained from the maximization of another objective function in a linear or non-linear way. The weights are then multiplied by the value of the drivers for each of the various subsidiaries to obtain actual allocations of the factors.

[12]   Another embodiment of the invention provides a method including using a computer system to provide projected tax liability when costs and revenue are allocated among a plurality of business entities, wherein the computer system executes a matching engine using weighting of attribute values.

[13]   Another embodiment of the invention provides a method including allocating cost or revenue among a plurality of business entities; using a computer system to provide projected tax liability based on the allocations; and using the computer system to provide incentive determination based on the allocations made in the step of allocating.

[14]   Another embodiment of the invention provides a method including allocating one of cost or revenue among a plurality of business entities; using a computer system to provide projected tax liability based on the allocations; and using the computer system to provide an audit risk projection based on the allocations made in the step of allocating.

BRIEF DESCRIPTION OF THE DRAWINGS

[15]   Fig. 1 is a diagram illustrating basic components of a system to analyze and optimize business operation.

DESCRIPTION OF THE SPECIFIC EMBODIMENTS

[16]   The invention is described in this application and in the documents accompanying this application named as follows:

[17]   (1) Liquid Engines Corporate Vision Brief, Draft October 15, 2001 (2 pages); and

[18]   (2) U.S. Patent Application Serial No. 09/823,955, filed March 30, 2001 entitled "ELECTRONIC MATCHING ENGINE FOR MATCHING DESIRED CHARACTERISTICS WITH ITEM ATTRIBUTES" (30 pages).

[19]   In Fig. 1, system 100 accepts factors such as cost 102 and revenue 104 as inputs. Note that other embodiments can accept other factors in addition to, or in place of, one or both of the cost and revenue factors. For example, sales regions, parts, or other resources necessary in creating a product or providing a service can also be "factors" allocated among subsidiaries or other business entities. In general, the system of the present invention is applicable to any tangible or intangible item, asset, service, agreement, instrument, resource or other characteristic of business operation that can be allocated among business entities. Factors can be of different types and can have sub-categories. For example, costs can be of fixed or variable types. Costs can be categorized as short-term, long-term, depreciable, etc. Costs can be categorized into utility (e.g., water, power), payroll, etc.

[20]   Allocation process 106 is symbolized by a rectangle. In a preferred embodiment, a user at a computer display can select and allocate factors. In this case, allocation 106 represents a manual operation that is typically performed by a suitable user interface such as graphical representation of factors on a computer screen, and allowing user selection and allocation with a user input device such as a mouse and pointer, trackball, touchscreen, etc. Factors can also be defined by a human user in relation to existing factors, values, metrics (discussed below), etc., or factors can be entered as new factors by suitably defining the factors. For example, where the matching engine described in the accompanying document is employed, factors can be defined as attribute/value pairs in mathematical, logical or relational equations, functions or algorithms.

[21]   Allocation process 106 can also be automated so that a computer system is used to, e.g., try different allocations to create a spectrum of predicted results. For example, visual graphs, maps, tables, etc., can be presented after many variations to a factor have been automatically performed. Such data can assist a human user to modify the factor, or other factors, to perform further analysis. As discussed below, automated allocation is also used when a human user directs the system to maximize, or optimize, specific metrics. Combinations of manual and automated allocation can be employed.

4

[22]   In a preferred embodiment, factors are allocated among business entities. In Fig.1, business entities are represented by circles within box 108.   Business entities can be any division, subsidiary, partnership, company, affiliate, member that is related to a business operation. The business entities need not all be within a same legal framework. In other words, it is possible for the system to be used in an analysis where factors are allocated among separate companies. Business entities can also simply be groups of people, corporations, or other physically, conceptually or legally defined entities that are involved with a business operation being analyzed by the system of the present invention.

[23]   In a preferred embodiment, the user can define business entities to suit particular types of analysis and optimization. Default groups of entities can exist in the form of templates, or other pre-defined collections. Not all of the business entities need have a factor allocated for accurate analysis and optimization to be achieved.

[24]   The analysis process is indicated by box 110. In a preferred embodiment, a specialized "matching engine" is used to perform the analysis. The matching engine is described in detail in the accompanying document entitled "ELECTRONIC MATCHING ENGINE FOR MATCHING DESIRED CHARACTERISTICS WITH ITEM ATTRIBUTES". The matching engine is well-suited to the type of analysis at issue in the present system. The matching engine includes variable weighting of attribute values. The weighting can be discrete or continuous. The matching engine can also selectively perform substitution for attributes, or other variables. These properties of the matching engine allow disparate types of values (e.g., money values and human performance measurements) to be manipulated and compared in meaningful ways without creating an undue processing burden. Although the present invention is discussed primarily in connection with the matching engine, it is possible to achieve, or use, aspects of the invention with other computing approaches.

[25]   The analysis process uses rules, or operations, describing a business' operating characteristics. These rules, and other data and functional code, are obtained from model 112. Model 112 can provide, for example, rules for deriving tax implications of business operation. A preferred embodiment of the present invention is being developed by Liquid Engines, Inc., as computer software referred to as the "Financial Allocation System." This product uses a model that includes descriptions of business agreements between business entities. For example, the model can include a rule that states "Entity A receives 40% of the operating budget of Entity B." Such agreements, or contracts, can be selectively employed by the user when running analyses using a specific model so that accurate, comprehensive

5

functioning of the business operations can be simulated. The agreements are described in a document referred to as the "360 degree financial record." In a tax implication model, tax-related cash flows specified in a contracts model can be used to optimize prices at which internal assets are traded. The cash flows, in turn, affect the ways in which objectives and incentives can affect business units, as described below.

[26]   The product uses five central modules that help determine optimized tax planning with respect to business economics. These modules include (1) Cost Allocation, i.e., the allocation of direct costs incurred by a corporation and its subsidiaries to its various entities; (2) Transfer Pricing, i.e., the amount that one subsidiary charges another for goods and services that benefit another subsidiary; (3) Income Shifting, i.e., the transferring of income from one entity or subsidiary to another; (4) Capital Investment, i.e., the effect of capital and other fixed expenditures on taxes and other financial metrics in the firm; and, (5) Financing Instruments, i.e., the choice of various methods (short term or long term debt, equity etc.) on taxes and other considerations for the firm. The system can apply rules, processes and data in these modules to perform analysis and optimization and make recommendations at any business entity whether geographical, divisional by product line, etc. This results in corporate-wide visibility to all financial data impacting the performance of business units, allowing the restructuring of transactions in a manner that enhances organization performance.

[27]   Note that many different models can be used to run different analysis. While the present invention is often discussed with respect to a tax analysis, any type of analysis is possible depending on the model, or models, used.

[28]   The result of analysis is the group of metrics at 120. These can be provided for a specific business entity, group of entities or for all entities. For example, tax liability metric 122 can be for the entire set of entities while audit risk 124 can be for a single, selected entity. Metrics are selected (automatically or manually) for use in subsequent allocation steps, as desired. Note that a subset of generated metrics can be used for another iteration of analysis. In Fig. 1, metrics of tax liability 122, audit risk 124 and incentive 126 are being used in subsequent analysis runs. Subsequent analysis can be automated by using conditional checks of the metric values, by using the metric values in calculations, etc. As an example, one analysis might be set up so factors are re-allocated and another analysis run as long as tax liability is over a given, threshold amount.

[29]   Audit risk metric 124 is a measure of the probability of a business entity (or group of entities) receiving a tax audit coupled with the expected cost associated with a particular

audit. The measure can be based on statistics obtained from records of actual audit frequency for different business practices or from subjective assessments by experts. The metric can include probability of reversal of an allocation and costs associated with an audit and reversal. The costs can include, but are not limited to, payback, penalties, interest and associated legal and accounting fees. The optimized solution can report audit risk and expected cost of a reversal in addition to calculating the optimal revenue and cost allocation strategy.

[30]   Incentive metric 126 is a measure of a business entity manager's incentive to keep costs down. For example, if a certain type of cost is not incurred by business unit A then the manager of business unit A has a lower incentive to keep costs down. Similar metrics can be developed for measuring other human tendencies that are important to business operations (e.g., sales, use of resources, etc.).

[31]   Although the invention has been described with respect to specific embodiments thereof, these embodiments are merely illustrative, not exclusive, of the invention. The scope of the invention is to be determined solely by the appended claims.

1   1.   A method for assigning a financial factor to one of a plurality of
2 business entities, the method comprising

3   identifying a factor to be assigned;

4   indicating a metric to be optimized;

5   defining one or more rules to be used in a matching process;

6   using a computer system to perform a step of comparing degrees of
7 optimization of the metric based on assignment of the factor to different ones of the plurality
8 of business entities, wherein the one or more rules are used to derive the degrees of
9 optimization; and

          using the result of the comparing step to assign the financial factor to the one
of a plurality of business entities.

          2.   The method of claim 1, wherein the factor includes cost.

          3.   The method of claim 1, wherein the factor includes revenue.

          4.   The method of claim 1, wherein the factor includes product shipping
allocation.

1   5.   The method of claim 1, wherein the factor includes sales allocation.

1   6.   The method of claim 1, wherein the business entities are located in
2 different geographic regions.

1   7.   The method of claim 1, wherein the business entities are located in
2 regions with different commercial laws.

1   8.   The method of claim 1, wherein the metric includes growth.

1   9.   The method of claim 1, wherein the metric includes profitability.

1   10.   The method of claim 1, wherein the metric includes tax liability.

1   11.   The method of claim 1, wherein the metric includes audit risk.

1   12.   The method of claim 1, wherein the metric includes return-on-assets

2 (ROA).

1   13.   The method of claim 1, wherein the metric includes human incentive.

1   14.   The method of claim 1, wherein the step of comparing degrees of

2 optimization uses a matching engine as described herein.

1   15.   A method for analyzing a business operation, the method comprising

2   using a computer system to provide projected tax liability when costs and

3 revenue are allocated among a plurality of business entities, wherein the computer system

executes a matching engine using weighting of attribute values.

                 16.   The method of claim 15, wherein the computer system uses discrete

weighting.

                 17.   The method of claim 15, wherein the computer system uses continuous

weighting.

                 18.   The method of claim 15, wherein the computer system uses

substitution of variables.

1   19.   A method for analyzing a business operation, the method comprising

2   allocating one of cost or revenue among a plurality of business entities;

3   using a computer system to provide projected tax liability based on the

4 allocations made in the step of allocating; and

5   using the computer system to provide incentive determination based on the

6 allocations made in the step of allocating.

1   20.   A method for analyzing a business operation, the method comprising

2   allocating one of cost or revenue among a plurality of business entities;

3   using a computer system to provide projected tax liability based on the

4 allocations made in the step of allocating;

5   using the computer system to provide an audit risk projection based on the

6 allocations made in the step of allocating.

COMPUTER-BASED OPTIMIZATION SYSTEM FOR FINANCIALS MANAGEMENT

ABSTRACT OF THE DISCLOSURE

A system for analyzing and optimizing allocation of factors among different business entities. The analysis, or optimization, can be targeted to a specific business financials measurement, or metric. Many different types of factors can be considered regardless of their initial definition, description or form. For example, along with revenues, profit, inventory size, etc., human characteristics such as incentive and performance can be measured and optimized. A preferred embodiment of the invention concerns analyzing factor allocation to reduce overall taxes for a company with subsidiaries in regions with different tax laws. The system includes consideration of local, state, federal and international taxes, transfer pricing, tax credit limitations, inter-state allocationsin unitary and non-unitary environments, carry-overs, and others. The system preferably uses a versatile matching engine, described herein, to automate the analysis and optimization. The matching engine is capable of discrete and continuous attribute value weighting, and of performing substitution of attributes, or other variables.
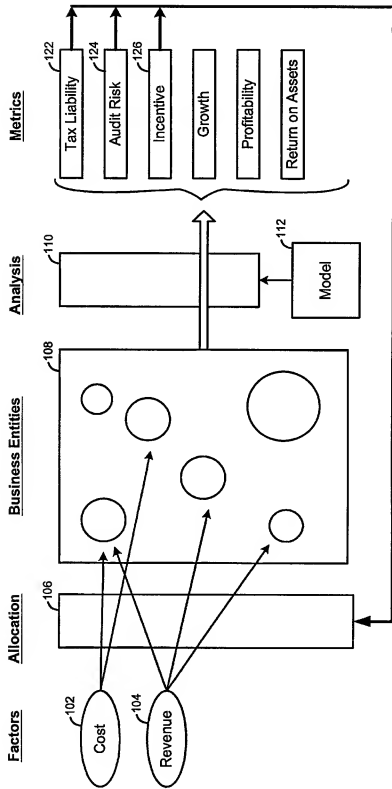
SF 1300874 v1

Factors   **Allocation**   **Business Entities**   **Analysis**   **Metrics**

Fig. 1

Liquid Engines
Corporate Vision Brief

Draft October 15, 2001

Mission

Liquid Engines, Inc. (LEI) is delivering enterprise financial visibility and optimization software that increases cash flows by synchronizing financial metrics with management incentives.

Business Economics Misalignment

Financial allocation has become an issue of strategic importance to global companies, as it directly impacts corporate and business-unit incentives and financial metrics like growth, profitability, effective tax rates and return-on-assets (ROA). Increasingly, global companies have recognized the need for more company-wide visibility on its forecasted cash flows. These flows are controlled by revenue, expense and capital decisions which are shared across geographic boundaries and need to be allocated across business units with respect to tax laws and GAAP methods.

Currently, the "typical" visibility and control of financial allocation for a global enterprise is highly problematic in a variety of ways:

> ➤ Financial personnel use individual, "homegrown" allocation models without corporate uniformity, resulting in significant time spent "negotiating" with business unit managers.
> ➤ Financial plans are done in "silos" by business unit managers, creating unintentional tax burdens and audit risks not caught early enough by the tax groups.
> ➤ Free cash flow is limited due to handcuffing corporate treasurers with inconsistent cross-geography tax schemes.
> ➤ Management incentives are misaligned without early tax burden visibility, resulting in 'hidden' enterprise churn and lost tax efficiencies.

The LEI Corporate-Wide Solution

The LEI Financial Allocation System provides CFOs the essential set of business processes and workflows for financial planning directors, business analysts, treasury managers, tax planner and divisional finance and business unit executive to have metrics and incentive visibility across the entire enterprise. The LEI system is an *economic-based* financial solution that acts on existing *operationally based* financial datastores, providing metrics optimization, variance alerts, causation reports and dispute resolution to continuously improve corporate-wide net cash flows.

The core document of the system is the *360 degree financial record*, which contains both explicit and implicit agreements of various parties to make decisions and to receive cash flows in differing circumstances. Specifically tax-related cash flows specified in these contracts can optimize prices at which internal assets are traded. These cash flows affect the ways in which objectives and incentives are organized by business units.

The LEI System is comprised of 5 central modules that help determine optimized tax planning with respect to business economics. These modules include: Cost allocation, Transfer pricing, Income shifting, Capital investment and Financing instruments. The system performs optimization, analysis and recommendation at any corporate node, whether geographical, divisional or by product line. The result is corporate-wide visibility to all financial data impacting the performance of business units, allowing the restructuring of transactions in a manner that enhances organization performance.

Attachment 1

Multi-Dimensional Optimization Technology

The LEI optimization technology is a first of its kind. Most optimization techniques are limited to solving 2-dimensional problems, and they have been applied in financial risk management and supply chain solutions with some success. None of these techniques can resolve the multi-attribute, multi-asset, multi-agent requirements for efficient business economic and tax decision making. There are 4 patents pending on the technology and business processes implemented in the LEI System.

The LEI Value Proposition

The Company is focused on delivering a self-funding ROI for its enterprise financial customers. There are two elements that comprise this value:
> *Increasing Cash Flow.* Utilizing the LEI system across all business units will provide minimally a 1% pre-tax relief on its operating income. This will mean $10s of millions of dollars annually to the bottom line, directly improving net cash flows.
> *Global visibility and control.* Centralizing the control and structure for enterprise allocations and incentives provides early visibility and cause/effect analysis, saving millions of dollars annually in business processes.

Why Now

Whether in up or down business cycles, a strong mandate exists for increasing cash flows and accountability in global enterprises. *Operationally based* financial systems, which gained popularity in the early 1990s, provided systematic ways of capturing and reporting on past data and metrics. CFOs are now clamoring for *economic-based* systems, providing analysis and optimization of forward financial outcomes. Technologically, web infrastructures and EAI platforms provide the ability to put these systems into the hands of all business functions with a non-disruptive integration process. Liquid Engines is capitalizing on these two major business and technology trends, leveraging its proprietary assets to create the first strategic enterprise cash flow management solution.

Background

Liquid Engines is an enterprise software company delivering a corporate-wide Financial Allocation System for aligning financial metrics with incentive compensation. The company began operations in the summer of 2000 and is located in Sunnyvale, CA. It was founded by Ms. Arti Arora and Dr. Ed Lazear, who co-developed core properties and algorithms that efficiently make complex business decisions using a multi-dimensional optimization technology. LEI's founders are at the forefront of a new economic practice, *Enterprise Cash Flow Management.*

LEI's customer is the global enterprise CFO. This executive has ultimate responsibility for all financial decisions and outcomes in a corporation. The LEI system gives the CFO, her/his employees and divisional managers the first solution to identify and optimize financial decisions that take into account both incentive and tax considerations. Inherent in this solution is a *new managerial approach*, one that focuses managers centrally on increasing and rewarding business accountability and cash flow on a continuous basis.

The company has assembled an employee and advisory team of world-renowned economists, tax policymakers and legal experts, financial systems knowledge-workers, and successful software entrepreneurs. In addition, the company has garnered strong financing from elite venture capitalists and will deliver it initial solution to the market in early 2002.

PATENT APPLICATION

ELECTRONIC MATCHING ENGINE FOR MATCHING DESIRED CHARACTERISTICS WITH ITEM ATTRIBUTES

Inventor(s):

Arti Arora, a citizen of India residing at
1221 St. Matthew Way
Los Altos, CA 94024

Edward Lazear, a citizen of the USA residing at
277 Old Spanish Trail
Portola Valley, Ca 94028

Assignee: Liquid Engines, Inc.
1289 Hammerwood
Sunnyvale, CA 94089

Entity: Small

Attachment 2

ELECTRONIC MATCHING ENGINE FOR MATCHING DESIRED CHARACTERISTICS WITH ITEM ATTRIBUTES

BACKGROUND OF THE INVENTION

20   This invention relates in general to digital processing and more specifically to a digital processing system for matching desired characteristics with item attributes.

One important use of electronic digital processing systems, such as computers, is to identify an item or object that is a "best match" with specified characteristics. Systems that perform such a matching function based on one or more criteria to identify a desired
25   choice, or choices, are called "matching engines."

An example of a matching engine is a general database query engine. A simple database query engine allows a user to specify one or more keywords and identifies documents containing the keywords. In such a system, a best match is often identified by the number of times the keywords appear in a document, the proximity of keywords to each other within a document, the placement of the keywords in different sections of a document (e.g., a
30   title), etc. Each document may be assigned a "score" or match value. A higher score can mean that the document is a better match to the query. A list of documents can be displayed

1

where earlier-listed documents have higher scores than later-listed documents so that the list is prioritized, making it easier for the user to identify a desired document.

More sophisticated matching engines are often used to create and facilitate "online markets." An example of an online market is an online auto dealership where a user is asked to specify a car to be purchased. For example, a user can enter characteristics for a desired car such as the type of car, color and age of the car. A user can specify the characteristics as a "new, silver sport utility vehicle." The matching engine will use the three desired characteristics of "new," "silver" and "sport utility vehicle" to compare against the attributes of item entries in a database. Only those entries that include attributes matching the specified characteristics will be returned.

The prior art matching process is not without shortcomings. The number and type of characteristics that can be entered by a user are typically defined by an administrator, or operator, of the site hosting the ecommerce engine, or marketplace. The user is often constrained to using all of the predefined characteristics. Another contraint is that only the predefined characteristics may be used. That is, the user can't specify any other characteristics other than those that the administrator has provided. This means that characteristics that are important to a buyer, such as airbags for example, may never enter into the matching process. Also, characteristics that are not important to a buyer may be required by the matching engine and might be used to eliminate items in which the buyer is actually interested. A second problem is that the prior art matching process is a one-to-one correspondence matching. If a certain color is specified by the buyer the matching engine does not take into account that other colors, or even other characteristics, of the car items may be satisfactory to a buyer. This drastically limits the number of suitable matches that will be identified and presented to buyers and, hence, reduces the number of sales and amount of revenue for the participants (i.e., buyers, sellers and administrating entity) of the online marketplace.

For example, in a prior art matching engine where the buyer must answer 20 "yes/no" questions to define the desired characteristics the chances of a direct match with an item's attributes are 1 in 2^{20} or about 1 in a million. This means that, for most applications, the number of matches will be very small, or none.

Another example is where a company has a catalog of goods having different attributes. Even companies with small catalogs (hundreds of items) can experience problems with prior art matching engines. For instance, a company may have approximately 400 types of shoes available on a website. Based on descriptions of the shoes, there may be 9 to 10

2

different attributes of shoes that customers care about. These attributes can include shoe style, usage (running, basketball, cross-training, etc.), price, etc. If buyers are asked to specify each of the 9 attributes as a characteristic where each characteristic has 3 choices this results in a database filter of approximately 20,000 possible combinations. Each shoe type has one set of attributes. In this case, there are 20,000 possible 'categories' but only 400 shoes. This implies that 19,600 categories are empty. With this matching engine approach there is a 98% chance that the search result will create no matches.

The way prior art matching engines alleviate this problem is by creating 'or' conditions within searches where a user can specify multiple values within each characteristic. 'Or' conditions can also be set up among characteristics so that all characteristics named by the user need to be present in a matching item's attributes. However, this moves the probability of a match to the other extreme. The obtains many more matches to the point where the results are not sufficiently targeted. For example, it is possible that 98% of all item entries show up as results in the search.

Thus, it is desirable to provide a matching engine that improves upon the shortcomings of the prior art and provides other advantages.

SUMMARY OF THE INVENTION

The present invention is a computer-based system for matching desired characteristics with item attributes. The system provides for weighting of variable values to be matched, and substitution of variables or values. Both discrete and continuous weighting can be used. This approach provides for more flexible matching to yield practical and useful results without placing high requirements on the computer system.

Weights can be assigned to variable values as defaults. Such assignment is usually performed by a system administrator, or the assignment can be calculated by a process in the matching engine (e.g., as a discrete or continuous function) or otherwise automatically derived. Weights can be selected by users (both buyers and sellers) by using a user interface that translates common expressions (e.g., "not required," "desired," "required") into weighting values between 0 and 1. Alternatively, users can assign weights as a number, or by other means.

One feature of a preferred embodiment of the invention allows preferences (i.e., characteristics and attributes) to be matched with regard to two different sides of a transaction. For example, both buyer and seller preferences can be taken into account in creating a match. This allows sellers to eliminate items or services from a particular

3

transaction based on seller goals of profitability, or where it makes a difference as to who the buyer is, or what is being offered in exchange for an item or service for sale. For example, in a job market system, the "seller" is an employer who may require prospective candidates to have a minimum number of years of education.

The matching engine can be applied to many types of applications. One envisioned application is to run an electronic marketplace such as an online store, auction, reverse auction, job placement center, etc. An electronic salesperson type of market is described that focuses on cost as a key factor in matching a buyer with an item for sale.

In one embodiment the invention provides a method for matching user preferences with item characteristics in an electronic database wherein items are stored in a database along with associated attributes and values, the method comprising accepting signals from the user input device to allow a user to specify preferences in the form of attributes and values; using the processor to identify one or more matches by using a weighted comparison among at least one value in the preferences and at least one value in the database.

In another embodiment the invention provides a method for matching user preferences with item characteristics in an electronic database, wherein items are stored in a database along with associated attributes and values, the database coupled to a processor and user input device, the method comprising accepting signals from the user input device to allow a user to specify preferences in the form of attributes and values; and using the processor to identify one or more matches after performing a step of substituting one or more attributes in the preferences.

BRIEF DESCRIPTION OF THE DRAWINGS

Fig. 1 illustrates a preferred embodiment system including the matching engine of the present invention.

DETAILED DESCRIPTION OF THE SPECIFIC EMBODIMENTS

The matching engine of the present invention can be applied to many applications where desired characteristics are used to determine best matches of items that are described using item attributes. A preferred embodiment of the invention creates and runs different types of online markets, such as an auction, reverse auction, exchange, etc. The preferred embodiment is incorporated into a suite of software products to be manufactured

4

and distributed by Liquid Engines, Inc. The suite of software is referred to as IXE2000 and is further described in the related patent applications, cited above.

Fig. 1 illustrates a preferred embodiment system of which the present invention is a part.

5 In Fig. 1, company 102 uses "configurator" software 104 to create an ecommerce engine 106. Ecommerce engine 106 is used to conduct electronic commerce in specific goods and/or services and in one or more market types. Ecommerce engine 106 includes methods and processing described herein for the matching engine. Configurator software 104 includes an administrator interface and market behavior data. An administrator

10 is a human operator who operates configurator software 104 and causes User Interface Generator software 108 to create user interfaces for buyers and sellers in different markets as shown by user interfaces 110, 112 and 114.

As buyers and sellers operate the user interfaces to characterize goods and services for sale and purchase (or trade), data is collected into databases 120 and 122 for

15 further use by the system. The data is accessed by ecommerce engine 106 for purposes of matching goods and services to buyers. Data such as transaction data is used in analysis, pricing and creating a stable market. In some situations, there are too few traders to create stable prices in a given market. The situation is highly volatile because any one buyer or seller can have a large effect on the equilibrium price. This causes problems since traders

20 will wait until a favorable time to trade and may even refuse to trade at a price that they previously stated would be acceptable. The situation arises most in new markets, where the case of a few traders is more common. The "ramp up" module uses a statistical approach to estimate the equilibrium price and to smooth out day-to-day variations that are not meaningful. Past data can be used to complement the limited information available as they

25 are ramping up so that the resulting matching and pricing information is meaningful and representative of the real word market.

Market behavior and user profiles are used by the User Interface generator to create dynamic user interface screens that are personalized to the specific exchange as well as the particular user that logs in. Intelligence algorithms work in conjunction with the

30 ecommerce engine, the various databases, and the configurator to recommend profitable, or desirable, markets to the administrator. The administrator can further model and analyze different potential markets, and can create and operate additional markets, as desired.

A preferred embodiment of the invention uses an Oracle database, XML (Extensible Markup Language), Hyper Text Markup Language (HTML) and Microsoft

5

Active Server Page (ASP) language, tools and applications to provide a system which executes on computers connected to the Internet. However, the features and functionality of the present invention can be implemented by many other suitable means such as with other computer programming languages, protocols, architectures or design methodologies. Any type of data communication can be used such as an intranet, local-area-network, point-to-point communications, etc. The processing and interfaces of the invention can operate on any digital processing platform such as desktop computers, servers, laptop or handheld computers or processors, etc.

A preferred embodiment is described in more detail in U.S. Patent Application Serial No._____ [TBA], filed March 30, 2001, entitled "Efficient Interface for Configuring an Electronic Market," referenced, above.

As previously mentioned, the matching engine described herein is used as part of ecommerce engine 106 of Fig. 1. In this preferred embodiment, the engine is a completely flexible, fully configurable piece of software that can run virtually any kind of market. All parameters can be set by the market administrator and modified at any time. The configurator allows the market administrator to set up the market in a short period of time without any technical expertise. The user interfaces generated by the configurator enable users to access the market by answering simple questions. The engine has default settings, but is completely customizable.

The matching engine allows for substitution of characteristics. A number of attributes are defined and each attribute is then described and valued, or "weighted," by a buyer, seller, or both. For example, an employer can specify that a desired characteristic of a recruit be that the recruit have a college degree. The employer is asked, by the interface, to specify how important the degree requirement is ranging from "essential" to "irrelevant." Note that the language and manner used to specify the weighting can vary and is typically set by the administrator. By using the characteristic's weight, or importance, the engine is able to generate matches with potential employees and arrive at a total score which is a function of all desired characteristics and weighting levels. Note that the use of weights can also be applied to item attributes. For example, a recruit can specify that a geographic location of San Francisco is an attribute of the recruit, but the weighting can be at 50% which can mean that a San Francisco location is "fairly" desirable as opposed to being a "mandatory" requirement.

Additionally, weighting values and schemes can be set by an administrator or computed or assigned by the engine, itself. For example, a default weighting can

6

automatically be assigned to attributes such as salary, or location. A function can be used to assign a weight to an attribute based on almost any type of criteria. An example is increasing the weighting of a salary attribute in relationship to a prospective employee's geographic proximity to a job's location that is specified as a desired characteristic.

5   Note that by using the matching engine of the present invention, both sides' (employers' and recruits') preferences can be taken into consideration. An advantage to this is that a potential employee does not need to be shown job positions which are desirable based on the potential employee's specified characteristics, but which are eliminated because of the employers' stated desired characteristics with respect to the recruit.

10   Matching in the present invention can be continuous in addition to being graduated, or "binary" (i.e., a match or no match). To use the above example, most cars neither match a buyer's desires perfectly nor not at all. Each potential buyer/seller match is valued. That match value becomes the basis on which all markets are run and all matches are made.

15   There is no limit to the number or types of attributes that can be used. The function allows for any finite number of characteristics. The probability of a match is not reduced by specifying more categories as would be the case in a prior art discrete matching approach.

      Many market forms that were previously impractical are made possible by the engine. The different types of market forms are discussed in the related applications,
20   referenced above.

      An example of the matching engine's weighting and substitution properties is next presented, followed by a mathematical description of the matching process.

25 Weighting and Substitution

      An example of an ecommerce market uses a matching of prospective car buyers' desired characteristics with dealers' car attributes stored in a database. An administrator designs a buyer interface that allows a buyer to define characteristics of color, type, age and other features of a desired car.

30   The characteristics can be input in a variety of ways. In a simplest format, the buyer merely answers "yes" or "no" to questions such as whether tilt steering, power windows, airbags, etc., must be present in the car. Other characteristics can be specified with a multiple-choice menu selection by the buyer such as the color of the car as "red," "blue,"

etc. Other characteristics may require a numerical input that can range continuously over many possible values such as the existing mileage of the car, and its horsepower.

Along with specifying the characteristics, the buyer is permitted to specify the importance of a characteristic. In a preferred embodiment, the importance value, or weight, of a characteristic, or attribute, ranges from 0 (ignore the characteristic) to 1 (the characteristic must be present in the match and must match exactly). For example, the buyer may assign an importance rating of 0.9 which could mean that the characteristic is very important but not essential.

Typically, the administrator designs a buyer interface so that a buyer need not be absolutely mindful of the weightings. This is accomplished by providing default, or automatically calculated, weights in some cases and by providing mappings of weights to common words or expressions in other cases. An example of an automatic weighting is a weighting of 0 given to those characteristics that the buyer omits, or fails to specify. For example, if the buyer does not reply "yes" or "no" to a "tilt_steering" characteristic specification then a weight of 0 can be assigned to the "tilt_steering" characteristic so that it is ignored in the matching process. Alternatively, a default non-zero importance, or weight, can be assigned to a characteristic that is omitted by the user. For example, the weight of the characteristic to an average user (statistically predefined) can be assigned when the weight is not specifically provided by the user. Another example of default weighting is where the buyer selects a value for a characteristic but fails to provide a weighting. For a color characteristic the default can be 0.3 when the buyer fails to specify a weighting since most buyers who fail to make color mandatory, or highly desirable, probably don't care too much about the color. For a car type characteristic a likely default would be 1 since car type (e.g., sedan, convertible) is usually an important, inflexible, characteristic for most buyers.

Other weightings can be calculated by the matching engine, or another process in the ecommerce system, based on a specified characteristic, multiple characteristic, or external data or factors. For example, the default weighting for a full-size spare tire can be higher where the buyer specifies an off-road vehicle type as opposed to an economy coupe.

The matching engine can act to translate, or modify, buyer-specified weightings. Typically a buyer will not be asked to input a numeric value in the range of 0-1 but can specify the weighting by using common expressions such as "not required," "not important," "desired," "very important" and "required." These expressions can be mapped to numeric weightings such as 0, 0.3, 0.5, 0.9 and 1, respectively.

8

The matching engine can substitute characteristics and attributes. For example, where a buyer specifies several safety devices such as front and side air bags, the engine can substitute an overall safety rating and give higher-rated cars higher scores for the safety-rating attribute. The engine can convert buyer-specified characteristics such as a "red" color into a numerical representation of the color such as a light wavelength, hue classification, etc., so that colors are matched on a continuous scale where dark orange colors result in a higher color attribute score than blue, or shorter-wavelength colors.

The matching engine can use discontinuous functions to perform attribute value matching such as where a buyer specifies that a car be "new." The matching engine can be configured to deem "new" cars as those cars with logged mileage under 100 or some other fixed number. "Average" and "old" cars ages can be defined where each definition is a range of miles. Alternatively, a continuous function can be used where the higher the number of miles, the lower the score for the "age" attribute, or characteristic. In general, the matching engine can use any type of function including discrete, continuous, limited set (e.g., based on alphabet values), etc., to describe variables and weightings. This eliminates the need to reduce the number of variables in a search to avaoid ending up with no matches. For example, where a typical automobile marketplace might ask a prospective buyer whether the desired auto should have "less than 20,000 miles" or greater than 20,000 miles, the present invention can accept the mileage as any number and use an appropriate continuous function to achieve matching. This is discussed in more detail, below.

The matching engine can accept characteristics from a buyer that are not pre-defined in the engine. For example, one type of user interface can allow a buyer to specify a keyword, value and weighting using alphanumeric characters such as "sunroof=yes 1" or "supercharger=yes 0.5" or "age=before 1935 1" – indicating the attribute, value and weighting. Similarly, the car dealer specifications of available cars – in the form of records in the ecommerce database – allow dealers to add attributes, values and weightings in a similar manner. If the same attribute names are used the attributes and characteristics can be matched in the same way as for pre-defined attributes.

Variable	Value buyer 1	Weight buyer 1	Value buyer 2	Weight buyer 2	Value seller 1	Weight seller 1	Value seller 2	Weight seller 2	Value seller 3	Weight seller 3
Color	Blue	.6	Red	1	Blue	0	Green	0	Black	0
Type	...									
Mileage	10000	.8	20000	.2	23147	0	10932	0	15578	0
Gas	...									

9

Variable	Value buyer 1	Weight buyer 1	Value buyer 2	Weight buyer 2	Value seller 1	Weight seller 1	Value seller 2	Weight seller 2	Value seller 3	Weight seller 3
mileage										
# doors										
Make										
Style										
Location (zip code)	94028	.9	94305	.2	94303	0	95129	0	94025	.1
Year										
…										
…										
…										
…										
…										
Date of delivery (desired if buyer or actual if seller)	Jan 3, 2001	.9	Feb 7, 2001	.8	Feb. 10	.9	Jan 2, 2001	.1	March 23, 2001	.9

TABLE I

The records in Table I illustrate the previously mentioned aspects of the matching engine.

In Table I, buyers 1 and 2 have a set of desired characteristics in a car that they want to purchase. Sellers 1, 2 and 3 (which may be different sellers or different cars of the same seller), also attach a weight to each characteristic (either verbally or numerically).

For example, buyer 1 wants a blue car and seller 1 has a blue car. Thus, they match perfectly on this attribute, getting a score of 1. Buyer 1 attaches a weight to this of .6. (E.g., he might have answered that color was "somewhat important" which the engine transforms according to previous instructions into a numerical value. Alternatively, the buyer could have merely reported a numerical value.) Color is a discrete variable in this example since it must be one of only a relatively few values. Note that seller's weight is zero, because he does not care about buyer's preferences over color, whereas the buyer cares about the actual color of the car.

Mileage is a continuous variable where less is better. Buyer 2 prefers a car with about 20,000 miles, but fewer miles are better and more are worse. This is a continuous

10

variable so a car with 10,000 miles receives a higher score on this attribute than one with 40,000 or even 20,000. The 20,000 response by the buyer benchmarks the function to be used in the tradeoff. Note also that buyer 2 is quite flexible on mileage, giving it a weight of only .2. Buyer 2 views Seller 2 best on this characteristic and seller 1 worst. Buyer 1 has the

5   same ranking as buyer 2, but places heavier weight on this characteristic. Note that the engine does not treat all cars with, say, less than 20,000 miles the same. Those with 10,000 are a better match than those with 18,000.

        Location is entered as a zip code. But the issue that it corresponds to is "how close is the dealership to the buyer's house." Thus, distance between buyer and seller is a

10   transformation of two zip codes, which depends on an absolute value (since distances are always positive). Buyer 1 cares a great deal about distance to the dealership; sellers 1 and 2 do not care about distance to the buyer. Seller 3 places a slight weight on distance to buyer, because that seller plans to deliver the car to the buyer's home.

        Finally, date of delivery is a continuous variable that is transformed from the

15   actual date desired. The closer to the desired date the better, but too early might be just as bad as too late because the buyer might not have the funds to purchase the car. Similarly, sellers have preferences over delivery dates as well. Sooner may be better because the seller does not have to cover holding costs. Later may be better because the seller will have more time to acquire and service the car. The ability to allow users to specify preferences is

20   provided by the administrator and user interfaces of the present invention as discussed in the related application, referenced above. Any specified preferences or characteristics can also have an attached weight, as well.

        The above examples illustrate the types of variables that can be accommodated. The approach of using weights with values can be generalized to any kind of

25   characteristic, attribute, property, preference, data, etc. that can be transformed into a value.

        The weighting function can handle any finite number of attributes, still providing non-zero matching values as long as a match does not violate an absolute restriction (e.g., when a non-matching value is discrete and its weight is 1). Conversely, no matter how good the match on, say, 999 of 1000 attributes, a score of zero results when at

30   least one party to the match says that there must be a perfect match on one discrete attribute and the attributes do not match.

        The quality of the match between buyer 1 and seller 1 depends on both the quality of the match on any given attribute and its importance. In particular, it satisfies the property that there can still be a good match when buyer and seller disagree on a

11

characteristic, but the characteristic is deemed to be irrelevant (or almost irrelevant) to the parties.

Using the example records in Table I, the matching engine compares characteristics values and weightings for each pair, triplet, or any number of agents with all other values in the database. A mathematical description of the matching process is provided in the next section below. Descriptive examples of the matching process are provided in the following paragraphs.

To achieve a total match score for any given party to a match, two records are chosen, e.g. the record of the worker who is shopping for a job and one of the many potential employers who are shopping for workers. Then the matching engine calculates the value of the match on each attribute, the weight on each attribute and then takes a linear or non-linear function of the attributes and weights to compute a score. That score in this example is the worker's view of the quality of the match with the seller in question.

The function that combines attributes and weights to obtain a score that rates the quality of the match to one of the parties can have any form. However, in the preferred functional form, given below, there is a linear and non-linear part combined with a power function that normalizes results.

After each party's view of the match is calculated, an overall match score for a pair of agents is computed as a function of each party's score. In the example above, there would be a score that relates the worker's view of the match with the employer in question. But there is also that employer's view of the match with the worker in question. Both are taken into account to get the overall match score. In this way, the algorithm allows for the value of a match to be a function of both the buyer's view of the seller as well as the seller's view of the buyer. Any function can be used to combine buyer score with seller score. In the preferred method described below, it is the square root of the product of the score for agent i and score for agent j, but it need not be restricted to that function.

Sometimes, one attribute makes sense for one side, but not for another. For example, a firm might care about a worker's education, but not the reverse. Then, the weight for the worker's view of that attribute is zero, and the firm's weight is between 0 and 1. Sometimes the reverse is true and sometimes both hold. For example, a person trying to buy a shoe on a web site cares about color, but not how much profit the firm earns on the sale of the shoe. The firm cares about profit, but not the color of the shoe since all shoe colors may cost the same to produce. In that case, the attributes that matter to the firm receive positive weights in the firm's scoring, but zero weights in the buyer's scoring. Those that are of

12

concern to the buyer receive positive weights in the buyer's scoring, but zero weights in the firm scoring. This can be done behind-the-scene as a default setting in the configurator or it can be entered explicitly by the users.

Default weightings, weighting substitution, weighting calculation as a function of elements in a database and user input and attribute substitution can all be dealt with. Furthermore, missing attributes can be dealt with in a number of ways. For example, a potential car purchaser might leave the "color' field blank. The weighting can then be set to zero (indicating that the user does not care about this attribute) or other rules can be used, e.g., assigning the mean or modal value to the attribute and the mean or modal weight.

The algorithm can report matches when a criterion level is met or can report all matches in order of their quality. The criterion level can be computed as a number or a function of other data in the database or can be specified directly by the users.

It should be apparent that the matching engine of the present invention can be used in various applications that would previously suffer from the shortcomings of prior art systems in handling "many-to-many" matching problems with many attributes. The matching engine of the present invention allows these problems to be solved and computed more quickly.

The matching engine can be used advantageously in diverse applications such as (1) assigning 1000 idiosyncratic consultants to 10,000 clients each with different preferences so as to maximize profits, overall value of matches or other criteria; (2) assigning flight attendants to flight slots; (3) assigning nurses with predefined qualifications and availability to patients with specified needs; (4) assigning resources to departments, etc.

Some of the features that the matching engine is capable of providing are shown in Table II. Various embodiments and variations of the matching engine can provide one or more (including all) of the features of Table II, below.

1.	If characteristic x is essential to agent A, then A views every match with a party not possessing characteristic x to be unacceptable.
2.	If characteristic x is more important to agent A than to B, then A values every match with a party possessing characteristic x to be no worse than B views it.
3.	If agent A cares more about characteristic x than agent B, then A views a match with another agent who possesses characteristic x as being no worse than agent B views that match (other factors constant).
4.	An agent who does not care about any attributes views a match with anyone as perfect.
5.	The value of a match must be able to depend on the value to both sides of the transaction.

	6.	An agent who cares about attributes does not necessarily view a match with an agent who does not care about any attributes as perfect.
	7.	False negatives should not be increased as the number of attributes increases.
5	8.	False positives should not increase in the number of attributes.
	9.	Continuous descriptors must be able to be handled.
	10.	Any functional form of attributes should be able to be handled.
	11.	Keywords should be able to be handled.
	12.	Any finite number of descriptors must be able to be handled.
10	13.	Many to many problems must be able to be handled.
	14.	If buyer A's characteristics are closer to those desired by the seller than buyer B's, then the seller's view of a match with A cannot be worse than the seller's view of a match with B.
15	15.	If seller A's characteristics are closer to those desired by the buyer than seller B's, then the buyer's view of a match with A cannot be worse than the buyer's view of a match with B.
	16.	If characteristic x is irrelevant to agent A, then changes in the value of x cannot affect the value of the match to agent A.

TABLE II

Mathematical Specification of the Engine

First, a set of attributes that are important in determining the value of a match between buyer and seller is defined. These attributes can take a variety of forms such as being discrete (e.g., in an occupation or not), continuous (e.g., level of education), an input to another variable (e.g., zip code), etc. From these attributes the key variables are formed in the model according to a number of possible methods, depending on the type of variable. The importance of each attribute is ranked on a scale from zero to 1 which can correspond to irrelevant to essential, completely flexible to completely inflexible, or other verbiage depending on the context of the variable in question. The importance is assigned to a_{i_r} meaning the importance that seller i attaches to attribute r or a_{j_r} meaning the importance that buyer j attaches to attribute r.

From the input variables, called X_{i_r} meaning the rth attribute for individual i (a seller) or X_{j_r} where j is the index for a buyer, D_{ij_r} are created which are variables that go from zero to 1, depending on how well a buyer's desires are satisfied by a seller's characteristics (or vice versa). The value that seller i attaches to a particular match is then

$$Z_{ij}^i = \sqrt{[\prod_r (1 - a_{i_r}(1 - D_{ij_r}))]^{\frac{1}{k}}[\sum_r \delta_{i_r}(1 - a_{i_r}(1 - D_{ij_r}))]}$$

14

where R is the total number of attributes indexed by r, D_{ijr} is the value of the match between seller i and buyer j on attribute r and

$$D_{ijr} = \max[0, (1 - \frac{C_{ijr}}{C^{max}_{r}})]$$

or

$$D_{ijr} = \max\{0, \min[(\frac{x_{ir} - C_{min}}{C^{bliss}_{r} - C_{min}}), 1]\}$$

or

$$D_{ijr} = \{0,1\}$$

or

$$D_{ijr} = \frac{\exp(b(x_{ir} - x_{jr})/\sigma_r)}{1 + \exp(b(x_{ir} - x_{jr})/\sigma_r)}$$

depending on the context. Here, C_{ijr} is a constructed variable based on a table look up or some other procedure. C_{ijr} is defined to be non-negative. Furthermore, the best value of C_{ijr} is zero. All positive values are worse. For example, C_{ijr} could be defined as distance between two

15

locations or C_{ijr} max is the max tolerable value. All values greater than this should give the calculated D_{ij} a value of zero.

In one form of D, the parameter b is set by the market administrator. For example, if b=1.946, then the logistic function shown is such that 2 standard deviations out gives a value of .98 or .02, depending on a negative or positive value.

Other functional forms for the calculation of D_{ijr} are shown below:

$$D_{ijr} = \max\{0, \min[1, 1 - \frac{|X_{ir} - X_{jr}|}{2 \cdot X_{half,r}}]\}$$

or

$$D_{ijr} = \max\{0, \min[1, 1 - \frac{1}{2}\left(\frac{X_{ir} - X_{jr}}{X_{half,r}}\right)^2]\}$$

or

$$D_{ijr} = \max\{0, \min[1, 1 - \frac{1}{2}\sqrt{\frac{|X_{ir} - X_{jr}|}{X_{half,r}}}]\}$$

where $X_{half,r}$ is the mean or median value of X for that attribute in the sample. These provide linear, concave and convex functions in X that have upper and lower supports.

Although the algorithm supports other possible functional forms for D, most of forms necessary for matching are described by those shown above. All this requires is that the market administrator is judicious in the choice and definitions of the X variables.

What can be done for seller i can also be done for buyer j so that a Z_{ij}^{i} can be defined as above. Then, the match value is

$$Z_{ij} = \left(Z_{ij}^{i} Z_{ij}^{j}\right)^{1/2}.$$

This method guarantees that there is always a best match. The quality of the match can be relatively good or relatively poor, but there is no doubt that a best match can always be found, simply defined as the highest Z_{ij} for any given seller i or buyer j. Unlike the categorical approach, increasing the number of variables does not diminish the probability of

5   finding a match. More variables means a more detailed description of what is a good match, but this may be improved or weakened by adding more attributes.

Once the match values are calculated, they can be used to put together markets. Examples of possible markets are described in the above-referenced related patent applications. Possible markets include Goods Exchanges, Service Exchanges, Competitive

10   Markets, Modified Competitive Markets, Consignment Stores, Barter, Electronic Pawnshops, Trading Posts, Qualified Auctions, Forward Contracts, Futures and Credit Markets, Electronic salesperson and Internal Allocation.

Properties of a specific ecommerce market, such as the price at which goods are sold, can be set in a variety of ways and are not specific to the engine. Each market can

15   use a variety of match algorithms and number of matches. For example, a buyer might want to see m sellers and a seller might want to see n buyers. Each match can be priced (so that the exchange gets a commission for each match), fees can be charged on the basis of actual trades that occur or a subscription fee can be charged.

Transaction data is entered into the ecommerce system. For example, where

20   the market is a job market, job seekers specify attributes and values regarding their qualifications, location, etc. A prospective employer can enter desired characteristics of candidates in the form of attribute/value pairs, also. The attribute/value (a/v) pairs can receive a weighting that is used in the matching engine process as described above.

Another example of a market is an exchange where User_1 is a provider of

25   item_A and seeks to obtain item_B. Both item_A and item_B are described using a/v pairs with optional weightings. User_1's item_A is matched to the desired items of other users in the market. Likewise, User_1's item_B is matched to provided items of other users in the market. The matching procedure can be designed to provide only the best single exchange between two users, or can be used to resolve (or "clear") item exchanges among all users, or

30   a subset of all users. For example, the ten best matches can be presented to the users for acceptance. The entity running the ecommerce marketplace can obtain a commission on completed exchanges.

In fact, the two agents need not be a buyer and seller at all. For example, the "buyer" could be a route that electricity could follow and the "seller" could be a particular

17

bundle of electrical power that is ready to be transmitted. The matching capability allows for any two or more agents or entities to be matched with one another.

Electronic Salesperson

5   Another type of market is a simple "salesperson" market where buyers provide desired characteristics of an item to be purchased. The desired characteristics are matched to item attributes. One important property of the salesperson market is that the buyer is allowed to state a price that the buyer is willing to pay. Price consideration is important since, unlike many other characteristics, it will often be the pivotal characteristic upon which the sale

10   hinges. The seller's position on price is taken into consideration so that sellers (or item providers) can be assured of obtaining an optimal, desired, or at least minimum profit.

The salesperson application is similar to the discussion, above, for Z_{ij}. Z_{ij}^i is the buyer side and is defined using the relevant characteristics including a D_{ij} for price, where D_{ij} price is defined as

$$D_{ijr} = \frac{\exp(1.946(x_{ir} - x_{jr})/\sigma_r)}{1 + \exp(1.946(x_{ir} - x_{jr})/\sigma_r)}$$

where x_{ij} is the price that the consumer states that he expects to pay, x_{jr} is the price of the article in question and σ_r is the standard deviation of the bid prices for consumers. On initial

20   setup, before any data are available, estimate σ_r by the standard deviation of the (unweighted or weighted by sales volume) selling prices of the items in the market.

The slight deviation is that on the seller side, Z_{ij}^j can be constructed as it normally is, but there is an alternative.

25   A firm wants to maximize expected profit by offering the good, j, that maximizes the following: max {(prob buys good_1)(profit on good_1), (prob buys good_2)(profit on good_2), ...(prob buys good_j)(profit on good_j)}.

Initially, Z_{ij} can be used as an estimate of square of prob buys 1. Later (see below) use market data to estimate logit where D_{ijr} and a_{ir} are right hand variables (using

30   functional form in the Z function). Then Z_{ij}^j is just the square of the profit made on good j.

18

Since everything will be ordinal, Z_{ij} can be normalized to between zero and one by defining

$$Z_{ij}^{\,j} = [(\delta_j - \delta_{min}) / (\delta_{max} - \delta_{min})]^2$$

where δ_j is the profit on good j, δ_{max} is the maximum profit on all goods in the set, and δ_{max} is the minimum profit on all goods in the set.

After the market is created, there will be data on buyers' purchases. With the purchase data, a logit is run where the r.h.s. variable is $Z_{ij}^{\,i}$. Then for next round, redefine

$$Z_{ij}^{\,i} \text{ as } Prob(Z_{ij}^{\,i})$$

where $Prob(Z_{ij}^{\,i})$ is the predicted probability from the logit, given Z_{ij}.

An alternative is to run the logit where the rhs variable is not $Z_{ij}^{\,i}$, but instead, the vector of

$$(1 - a_{i_t}(1 - D_{ijr}))$$

and to use the coefficients on the logit to get a predicted probability. Then the $Z_{ij}^{\,i}$ is simply the predicted probability of sale from the logit, where the a and D data are entered by the user. This is probably the better approach, although the fit of the logit (-2 log likelihood) will tell us.

The price should not be taken as given. Instead, marginal cost of the item should be given as data rather than the profit per item. Could solve the problem by calculating the profit maximizing price for each of the j items and then offer the one at the price (e.g., list minus the calculated discount) that maximizes profit across all items.

A numerical approximation should work quite well. Take list price on good j. Then calculate (list price - marginal cost). Calculate the probability of a sale for good j based on prices that vary from list price down to marginal cost in h increments, i.e.,

prob of sale as function of attributes and price where

19

price = list price - (g/h)(list price - marginal cost)   for g = 0 to h

(actually could do it for g = 0 to h-1 because g=h yields zero profit).

5   Then pick the price for good j that maximizes (prob of sale)(price - marginal cost). Do this across all goods j and show the highest profit good or goods in order of their expected profit with discounts offered so that the purchase price is the optimal price for that good.

Note that an analagous type of computation can be performed for the
10   advantage of the buyer. The buyer is typically not interested in profit margin – but only in ultimate price. In the buyer case, prices of all goods j are compared while taking into account discounts, regional taxes, promotional offerings, etc., to minimize the ultimate price.

A preferred embodiment shows up to 3 offerings. The offerings can be made dynamic so that the displayed offerings, or possible buyer choices, are changed periodically
15   based on real-time gathering and computation of market data. If changes in market data create price differences among the offerings that disfavor the seller (or, alternatively, disfavor the buyer) then offerings can be removed (or added). For example, if the expected profit from best to second best choice falls off too dramatically, i.e., exceeds some critical value (or function), then only one choice can be shown. Similarly, the same approach can be taken
20   among all offerings such as between offerings 2 and 3, etc.

Note that the present invention allows sellers to "push" products that they prefer to sell. For example, if a buyer prefers item A over item B, but item B's sale would yield a higher profit for a seller, the matching engine can be set to offer item B and not item A.

25   Although the invention has been described with respect to specific embodiments thereof, these embodiments are merely illustrative, not exclusive, of the invention. For example, the matching engine has been discussed by using the terms "desired characteristics" and "item attributes." However, these terms are functionally equivalent so that the notions of "characteristics" and "attributes" can be interchanged throughout the
30   discussion, herein. The term "characteristics" has been used to indicate attributes that a user specifies to search for a match among stored items in a database. The term attributes has been used to describe traits of items to match with the characteristics. However, the engine can match stored item attributes with stored item attributes (as where two databases of items

are compared); specified characteristics with specified characteristics (as where two groups of users indicate preferences and the preferences are matched), etc.

The principles of price and offering in the "Electronic Salesperson" section can be applied to markets other than the "salesperson" market. All of the various features and computations of the invention need not be present in a given system. Also, additional features can be employed to supplement the features presented herein without departing from the scope of the invention.

Any manner of hardware and software can be employed to achieve the present invention. Although a preferred embodiment of the invention uses client computers coupled to one or more server computers via the Internet, other approaches include using a local-area network or standalone system. A dedicated network can be used, or a single machine can be used to provide all of the processing and user interfaces. For example, a multi-user time-shared environment where users access a single computing machine can be used. Other approaches are possible.

The matching engine and associated components and processing can be distributed over several digital processing, or digital handling, hardware components and software processes. The design of hardware and software can vary widely.

Many techniques can be employed to identify matches, assign weightings, interpolate weightings, etc. For example, complementary slackness approaches, such as epsilon complementary slackness, can be used to assist in determining matches. Note that not all of the techniques and steps described herein need be employed in any given matching engine. A subset of the steps, features, or characteristics of the matching engine of the present invention can be employed, as desired. Additional steps, or processing, can be used in conjunction with those presented, herein.

Thus, the scope of the invention is to be determined solely by the appended claims.

1   1.   A method for matching user preferences with item
2 characteristics in an electronic database, wherein items are stored in a database along
3 with associated attributes and values, the database coupled to a processor and user
4 input device, the method comprising
5   accepting signals from the user input device to allow a user to specify
6 preferences in the form of attributes and values;
7   using the processor to identify one or more matches by using a
8 weighted comparison among at least one value in the preferences and at least one
9 value in the database.

     2.   The method of claim 1, further comprising.
     using a continuous weighting comparison.

     3.   The method of claim 1, further comprising.
     using a discontinuous weighting comparison.

     4.   The method of claim 1, further comprising.
     using a linear weighting comparison.

     5.   The method of claim 1, further comprising
     using a non-linear weighting comparison.

1   6.   A method for matching user preferences with item
2 characteristics in an electronic database, wherein items are stored in a database along
3 with associated attributes and values, the database coupled to a processor and user
4 input device, the method comprising
5   accepting signals from the user input device to allow a user to specify
6 preferences in the form of attributes and values;
7   using the processor to identify one or more user matches by using a
8 weighted comparison among at least one value in the preferences and at least one
9 value in the database;

10   using the processor to identify one or more item matches by using a

11   weighted comparison among at least one value in the database and at least one value

12   in the preferences; and

13   informing the user of best matches, wherein the best matches include at

14   least one match from the one or more user matches and at least one match from the

15   one or more item matches.

1   7.   The method of claim 1, wherein the step of using the processor

2   to identify one or more matches includes substeps of

3   identifying matches by deriving a value from the weighted comparison,

4   wherein values indicate a range of matches from strong to weak; and

5   using a condition to identify a match, wherein the condition results in a

6   match being identified where the match is not the strongest match.

           8.°   The method of claim 7, wherein the condition includes a

2   consideration of a profit margin in completing a transaction based on the identified

3   match, the method further comprising

4   selecting the match that results in the highest profit margin.

1   9.   The method of claim 1, further comprising

2   indicating the one or more matches to a user.

1   10.   The method of claim 1, further comprising

2   initiating a transaction based on the one or more matches.

1   11.   The method of claim 1, wherein an attribute has multiple

2   selections per attribute.

1   12.   The method of claim 1, wherein items are goods.

1   13.   The method of claim 1, wherein items are services.

1   14.   A method for matching buyer preferences with characteristics

2   of items being sold in an electronic marketplace, the method comprising

3   accepting input from buyers to define buyer preferences as

4   attribute/value pairs;

5   storing definitions of items as attribute/value pairs; and

6   using weighting information with the attribute/value pairs to match

7 buyer preferences with item characteristics by deriving a score for a match.

1   15. The method of claim 14, wherein different weighting

2 information is associated with two or more item characteristics.

1   16. The method of claim 14, wherein different weighting

2 information is associated with two or more buyer preferences.

1   17. A method for generating a score for the strength of a match

2 between first and second sets of attribute/value pairs, the method comprising

3   deriving a first score to indicate the strength of a match of the first set

4 to the second set; and

5   deriving a second score to indicate the strength of a match of the

6 second set to the first set, wherein the first and second scores are not the same.

1   18. The method of claim 1, wherein an attribute is the time at

2 which an event occurs.

1   19. The method of claim 17, wherein the time attribute has a range

2 of values.

1   20. The method of claim 1, wherein a location attribute is used to

2 indicate location, the method further comprising

3   computing the absolute difference between locations specified by first

4 and second location attributes;

5   using the absolute difference in identifying one or more matches.

1   21. The method of claim 1, wherein attributes can have continuous

2 values.

1   22. The method of claim 20, wherein an attribute represents

2 education in years.

1   23. The method of claim 20, wherein an attribute represents size.

1   24. The method of claim 20, wherein an attribute represents

2 weight.

1   25. The method of claim 1, further comprising

25

2   transforming a first value associated with a first attribute into a second

3   value associated with the first attribute, wherein the second value is used in place of

4   the first value to identify one or more matches by using a weighted comparison.

1   26.   The method of claim 1, wherein the user designates multiple

2   attributes and values to specify a preference.

1   27.   The method of claim 1, wherein the step of using the processor

2   to identify one or more matches includes the step of

3   using epsilon complementary slackness to identify the one or more

4   matches.

1   28.   The method of claim 1, further comprising

2   performing a subsequent matching operation after removing

3   preferences and characteristics of the one or more identified matches.

1   29.   A method for matching user preferences with item

2   characteristics in an electronic database, wherein items are stored in a database along

3   with associated attributes and values, the database coupled to a processor and user

4   input device, the method comprising

5   accepting signals from the user input device to allow a user to specify

6   preferences in the form of attributes and values;

7   using the processor to identify one or more matches after performing a

8   step of

9   substituting one or more attributes in the preferences.

1   30.   A method for matching user preferences with item

2   characteristics in an electronic database, wherein items are stored in a database along

3   with associated attributes and values, the database coupled to a processor and user

4   input device, the method comprising

5   accepting signals from the user input device to allow a user to specify

6   preferences in the form of attributes and values;

7   using the processor to identify one or more matches after performing a

8   step of

9   substituting one or more attributes in the characteristics.

1   31.   A method for matching user preferences with item

2 characteristics in an electronic database, wherein items are stored in a database along

3 with associated attributes and values, the database coupled to a processor and user

4 input device, the method comprising

5   accepting signals from the user input device to allow a user to specify

6 preferences in the form of attributes and values;

7   substituting one or more attributes in either the characteristics or the

8 preferences; and

9   subsequent to the step of substituting, performing the step of using the

10 processor to identify one or more matches by using a weighted comparison among at

11 least one value in the preferences and at least one value in the database.

ELECTRONIC MATCHING ENGINE FOR MATCHING DESIRED CHARACTERISTICS WITH ITEM ATTRIBUTES

ABSTRACT OF THE DISCLOSURE

A digital system for matching desired characteristics with item attributes. The system provides for weighting of variable values to be matched, and substitution of variables or values. Both discrete and continuous weighting can be used. This approach provides for more flexible matching to yield practical and useful results without placing high requirements on the computer system. Weights can be assigned to variable values as defaults. Such assignment is usually performed by a system administrator, or the assignment can be calculated by a process in the matching engine (e.g., as a discrete or continuous function) or otherwise automatically derived. Weights can be selected by users (both buyers and sellers) by using a user interface that translates common expressions (e.g., "not required," "desired," "required") into weighting values between 0 and 1. Alternatively, users can assign weights as a number, or by other means. One feature of a preferred embodiment of the invention allows preferences (i.e., characteristics and attributes) to be matched with regard to two different sides of a transaction. For example, both buyer and seller preferences can be taken into account in creating a match. This allows sellers to eliminate items or services from a particular transaction based on seller goals of profitability, or where it makes a difference as to who the buyer is, or what is being offered in exchange for an item or service for sale. For example, in a job market system, the "seller" is an employer who may require prospective candidates to have a minimum number of years of education.

Fig. 1

CUSTOMERS
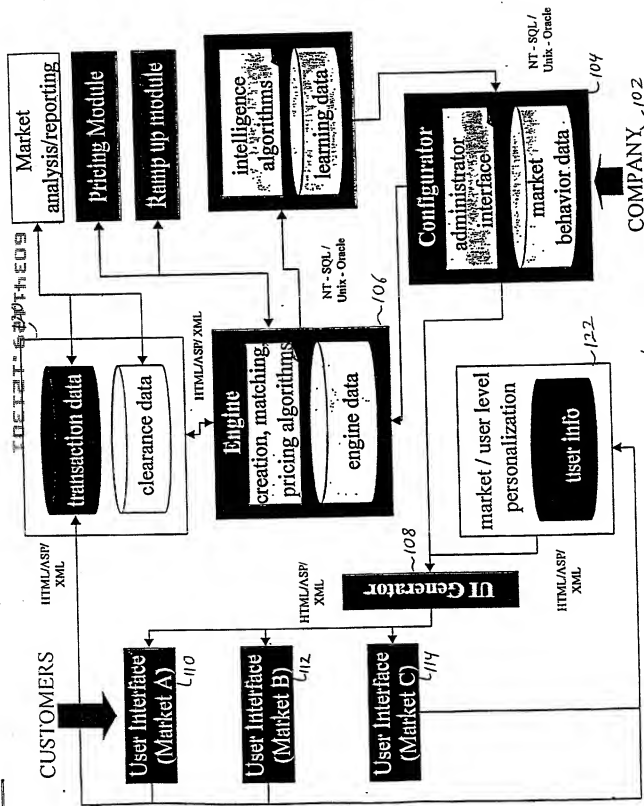
COMPANY ~102

Market analysis/reporting

Pricing Module

Ramp up module

intelligence algorithms / learning data

Configurator
administrator interface
market behavior data

NT - SQL / Unix - Oracle ~104

transaction data

clearance data

Engine
creation, matching pricing algorithms
engine data

NT - SQL / Unix - Oracle ~106

market / user level personalization
user info ~122

UI Generator ~108

User Interface (Market A) ~110

User Interface (Market B) ~112

User Interface (Market C) ~114

HTML/ASP/ XML

IDE121 9:20141 EO9